IN THE CLAIMS

The following is a complete listing of the claims, and replaces all earlier versions and listings.

1. (Previously Presented) A method of entropy coding of discrete wavelet transform coefficient bits that are arranged in code blocks and coded in bitplane order using three coding passes for each bitplane, said method including the steps of:

pre-analyzing transform coefficients of a code block in sign-magnitude form to determine statistical data about the coefficients;

storing the statistical data; and

generating, based upon the statistical data, a command for arithmetic encoding of a sequence, wherein the sequence comprises a plurality of bit and context pairs for the code-block, and wherein the command spoecifies a length of the sequence.

2. (Previously Presented) The method according to claim 1, wherein the statistical data is stored with the coefficients.

3. (Previously Presented) The method according to claim 1, further including the step of buffering significance state data, coded data, magnitude refinement data, bit data, and sign data for the code block.

4. (Original) The method according to claim 3, wherein said buffering step is implemented using register arrays for context generation.

5. (Previously Presented) The method according to claim 4, further including the step of switching to a specific region of the code block in any one of significance propagation, magnitude refinement, and cleanup coding passes.

6. (Previously Presented) The method according to claim 5, wherein said switching step is implemented using rotate-left and rotate-up operations of the register arrays.

7. (Previously Presented) The method according to claim 1, further including the step of buffering bit and context data before arithmetic coding using the bit and context data.

8. (Previously Presented) The method according to claim 7, wherein bit, context and the number of bit and context pairs are buffered.

9. (Original) The method according to claim 8, wherein said buffering step is implemented using a FIFO.

10. (Previously Presented) The method according to claim 1, further including the steps of:

buffering a region of a code block, the region being currently coded; and

buffering separately the remaining regions of the code block.

11. (Currently Amended) The method according to claim 10, wherein:

said buffering step for the region being currently coded is implemented using a register window; and

said buffering ~~stop~~ step for the remaining regions is implemented using a scratch memory.

12. (Previously Presented) The method according to claim 1, further including the steps of:

looking for, using a bypass control module, the next region of a code block to be coded in each of significance propagation, magnitude refinement, and cleanup coding passes; and

generating, using a context generation module, a context of a region previously provided by the bypass control module, the bypass control and context generation modules operating in parallel.

13. (Previously Presented) The method according to claim 12, further including the step of communicating data between the bypass control and context

generation modules using a bus, the bus including a control bus and a data bus, the control bus providing an indication of which column to start in a region when the context generation module processes the region.

14. (Previously Presented) The method according to claim 13, further including the step of detecting termination of processing in the context generation module for each pass, the detected termination enabling coding in the region to be terminated before scanning to a last column.

15. (Previously Presented) The method according to claim 1, wherein the entropy coding is JPEG 2000 entropy coding.

16. (Previously Presented) An apparatus for entropy coding of discrete wavelet transform coefficient bits that are arranged in code blocks and coded in bitplane order using three coding passes for each bitplane, said apparatus including:

means for pre-analyzing transform coefficients of a code block in sign-magnitude form to determine statistical data about the coefficients;

means for storing the statistical data; and

means for generating, based upon the statistical data, a command for arithmetic encoding of a sequence, wherein the sequence comprises a plurality of bit and context pairs for the code-block, and wherein the command specifies a length of the sequence.

17. (Previously Presented) The apparatus according to claim 16, wherein the statistical data is stored with the coefficients.

18. (Previously Presented) The apparatus according to claim 16, further including means for buffering significance state data, coded data, magnitude refinement data, bit data, and sign data for the code block.

19. (Previously Presented) The apparatus according to claim 18, wherein said buffering means comprises register arrays for context generation.

20. (Previously Presented) The apparatus according to claim 19, further including means for switching to a specific region of the code block in any one of significance propagation, magnitude refinement, and cleanup coding passes.

21. (Previously Presented) The apparatus according to claim 20, wherein said switching means implements rotate-left and rotate-up operations of the register arrays.

22. (Previously Presented) The apparatus according to claim 16, further including means for buffering bit and context data before arithmetic coding using the bit and context data.

23. (Previously Presented) The apparatus according to claim 22, wherein bit, context and the number of bit and context pairs are buffered.

24. (Original) The apparatus according to claim 23, wherein said buffering means is a FIFO.

25. (Previously Presented) The apparatus according to claim 16, further including:

means for buffering a region of a code block, the region being currently coded; and

means for buffering separately the remaining regions of the code block.

26. (Currently Amended) The apparatus according to claim 25, wherein:

said buffering means for the region being currently coded is a register window; and

said buffering means for the remaining regions is implemented using a scratch memory.

27. (Previously Presented) The apparatus according to claim 16, further including:

a bypass control module for looking for the next region of a code block to be coded in each of significance propagation, magnitude refinement, and cleanup coding passes; and

a context generation module for generating a context of a region previously provided by the bypass control module, the bypass control and context generation modules operating in parallel.

28. (Previously Presented) The apparatus according to claim 27, further including a bus for communicating data between the bypass control and context generation modules, the bus including a control bus and a data bus, the control bus providing an indication of which column to start in a region when the context generation module processes the region.

29. (Previously Presented) The apparatus according to claim 28, further including means for detecting termination of processing in the context generation module for each pass, the detected termination enabling coding in the region to be terminated before scanning to a last column.

30. (Previously Presented) The apparatus according to claim 16, wherein the entropy coding is JPEG 2000 entropy coding.

31. (Previously Presented) A computer program product having a computer readable medium having a computer program recorded therein for entropy coding of discrete wavelet transform coefficient bits that are arranged in code blocks and coded in bitplane order using three coding passes for each bitplane, said computer program product including:

computer program code means for pre-analyzing transform coefficients of a code block in sign-magnitude form to determine statistical data about the coefficients;

computer program code means for storing the statistical data; and

computer program code means for generating, based upon the statistical data, a command for arithmetic encoding of a sequence, wherein the sequence comprises a plurality of bit and context pairs for the code-block and wherein the command specifies a length of the sequence.

32. (Previously Presented) The computer program product according to claim 31, wherein the statistical data is stored with the coefficients.

33. (Previously Presented) The computer program product according to claim 31, further including computer program code means for buffering significance state data, coded data, magnitude refinement data, bit data, and sign data for the code block.

34. (Original) The computer program product according to claim 33, wherein said buffering computer program code means implements register arrays for context generation.

35. (Previously Presented) The computer program product according to claim 34, further including computer program code means for switching to a specific region of the code block in any one of significance propagation, magnitude refinement, and cleanup coding passes.

36. (Previously Presented) The computer program product according to claim 35, wherein said computer program code means for switching implements rotate-left and rotate-up operations of the register arrays.

37. (Previously Presented) The computer program product according to claim 31, further including computer program code means for buffering bit and context data before arithmetic coding using the bit and context data.

38. (Previously Presented) The computer program product according to claim 37, wherein bit, context and the number of bit and context pairs are buffered.

39. (Original) The computer program product according to claim 38, wherein said buffering computer program code means implements a FIFO.

40. (Previously Presented) The computer program product according to claim 31, further including:

computer program code means for buffering a region of a code block, the region being currently coded; and

computer program code means for buffering separately the remaining regions of the code block.

41. (Currently Amended) The computer program product according to claim 40, wherein:

said computer program code means for buffering the region being currently coded is a register window; and

said buffering means for the remaining regions is a scratch memory.

42. (Previously Presented) The computer program product according to claim 31, further including:

computer program code means for implementing a bypass control module to look for the next region of a code block to be coded in each of significance propagation, magnitude refinement, and cleanup coding passes; and

computer program code means for context generation module to generate a context of a region previously provided by the bypass control module, the bypass control and context generation modules operating in parallel.

43. (Previously Presented) The computer program product according to claim 42, further including a bus for communicating data between the bypass control and context generation modules, the bus including a control bus and a data bus, the control bus providing an indication of which column to start in a region when the context generation module processes the region.

44. (Previously Presented) The computer program product according to claim 43, further including computer program code means for detecting termination of processing in the context generation module for each pass, the detected termination enabling coding in the region to be terminated before scanning to a last column.

45. (Previously Presented) The computer program product according to claim 31, wherein the entropy coding is JPEG 2000 entropy coding.

46.-77. (Canceled)

78. (Previously Presented) A method according to claim 1, wherein the statistical data comprises a predicted significance state after a first cleanup coding pass.